

SSSSSSSS SSSSSSSS YY YY SSSSSSSS SSSSSSSS EEEEEEEEEE RRRRRRRR RRRRRRRR AAAAAAA AAAAAAA PPPPPP PPPPPP PPPPPP PPPPPP AAAA AAAA TTTTTTTT TTTTTTTT
 SSSSSSSS SSSSSSSS YY YY SS SS EE EE RR RR AA AA AA AA PP PP AA AA AA AA AA TT TT
 SS SS YY YY SS SS EE EE RR RR AA AA AA AA PP PP AA AA AA AA AA TT TT
 SS SS YY YY SS SS EE EE RR RR AA AA AA AA PP PP AA AA AA AA AA TT TT
 SSSSSS SSSSSS YY YY SSSSSS SSSSSS EEEEEEEE RRRRRRRR RRRRRRRR AA AA AA AA PPPPPP PPPPPP PPPPPP PPPPPP AAAA AAAA AA AA TT TT
 SSSSSS SS YY YY SS SS EE EE RR RR RR RR AA AA AA AA AA PP PP AAAA AAAA AA AA TT TT
 SS YY YY SS SS EE EE RR RR RR RR AA AA AA AA AA PP PP AAAA AAAA AA AA TT TT
 SS YY YY SS SS EE EE RR RR RR RR AA AA AA AA AA PP PP AAAA AAAA AA AA TT TT
 SSSSSSSS SSSSSSSS YY YY SSSSSSSS SSSSSSSS EEEEEEEEEE RRRRRRRR RRRRRRRR AA AA AA AA PP PP AA AA AA AA TT TT
 SSSSSSSS SSSSSSSS YY YY SSSSSSSS SSSSSSSS EEEEEEEEEE RRRRRRRR RRRRRRRR AA AA AA AA PP PP AA AA AA AA TT TT

(1)	57	Declarations
(1)	72	Entry vector
(1)	107	Main routine

```
0000 1 .TITLE SYSERAPAT - Generate a security erase pattern
0000 2 .IDENT 'V04-000'
0000 3 :*****
0000 4 :*****
0000 5 :*
0000 6 :* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 7 :* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 8 :* ALL RIGHTS RESERVED.
0000 9 :*
0000 10 :* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 11 :* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 12 :* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 13 :* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 14 :* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 15 :* TRANSFERRED.
0000 16 :*
0000 17 :* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 18 :* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 19 :* CORPORATION.
0000 20 :*
0000 21 :* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 22 :* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 23 :*
0000 24 :*
0000 25 :*****
0000 26 :*
0000 27 :++
0000 28 :* FACILITY: VMS Executive, System services.
0000 29 :*
0000 30 :* ABSTRACT:
0000 31 :*
0000 32 :* Generate and return a security erase pattern. This code
0000 33 :* is more or less a place holder for a user written routine to
0000 34 :* accomplish the same function. The erase pattern returned by
0000 35 :* this routine will always be zero.
0000 36 :*
0000 37 :*
0000 38 :* ENVIRONMENT:
0000 39 :*
0000 40 :* Kernel Mode
0000 41 :*
0000 42 :* AUTHOR:
0000 43 :*
0000 44 :* Steven T. Jeffreys
0000 45 :*
0000 46 :* CREATION DATE:
0000 47 :*
0000 48 :* 24-September-1982
0000 49 :*
0000 50 :* MODIFIED BY:
0000 51 :*
0000 52 :* V03-001 STJ3054 Steven T. Jeffreys,
0000 53 :* Removed EXESERAPAT_DEF definition. 21-Jan-1983
0000 54 :*-
0000 55 :*
```

0000 57 .SBTTL Declarations
0000 58 \$ERADEF : Define erase type codes
0000 59 \$PSLDEF : Define PSL fields
0000 60 \$\$SDEF : Define status codes
0000 61
0000 62 :
0000 63 : Equated symbols:
0000 64 :
0000 65
00000004 0000 66 TYPE = 4 : Offset to TYPE parameter (value)
00000008 0000 67 COUNT = 8 : Offset to COUNT parameter (value)
0000000C 0000 68 PATADR = 12 : Offset to PATADR parameter (address)
00000001 0000 69 MAXCOUNT= 1 : Maximum count (erase 1 time)
0000 70

0000 72 .SBTTL Entry vector
0000 73 :+
0000 74 : The following vectors are used by the various pieces of the system
0000 75 : to access the erase pattern generator. The vector EXE\$SERAPAT is
0000 76 : used by the change mode dispatcher in response to a user calling the
0000 77 : SERAPAT system service. This vector then jumps to the actual dispatch
0000 78 : vector, EXE\$SERAPAT_VEC, which in turn will jump to erase pattern
0000 79 : generator code. This level of indirection is necessary because the
0000 80 : change mode dispatch vector must be in close proximity to the change
0000 81 : mode dispatcher, which implies that it must be in a read-only psect.
0000 82 : The actual dispatch vector, EXE\$SERAPAT_VEC, must be in a writable
0000 83 : psect so that the contents of the vector may be changed.
0000 84 :
0000 85 : The longword SGNSGL_LOADFLAGS is a bit vector used to indicate which
0000 86 : pieces of the loadable pieces of the EXEC should be loaded at system
0000 87 : boot time. If a user specified erase pattern generator routine is
0000 88 : present in the system, the bit SGNSV_LOADERAPT will be set to 1.
0000 89 : This fact can be used to the advantage of the EXEC to avoid the overhead
0000 90 : of having to call the default erase pattern generator, since it always
0000 91 : returns a zero, and is a one-step erase function.
0000 92 :
0000 93 : The vector address the user must specify to load the code is represented
0000 94 : by the symbol EXE\$SERAPAT_VEC.
0000 95 :
0000 96 :-
0000 97 :
0000 98 .PSECT AEXENONPAGED : Nonpaged UR access only
0000 99 EXE\$SERAPAT:: : Entry point from change-mode disp.
0000 100 .WORD 0 : Register save mask (none saved)
0000 101 JMP @#EXE\$SERAPAT_VEC : Jump to the dispatch vector
0000 102 :
0000 103 .PSECT \$SS\$500 : The vector must be nonpaged and URKW
0000 104 EXE\$SERAPAT_VEC:: : Quick access entry point
0000 105 JMP @#EXE\$SERAPAT_RTN : Vector to default routine
0000 106 :
0000 107 :
0000 108 :
0000 109 :
0000 110 :
0000 111 :
0000 112 :
0000 113 :
0000 114 :
0000 115 :
0000 116 :
0000 117 :
0000 118 :
0000 119 :
0000 120 :
0000 121 :
0000 122 :
0000 123 :
0000 124 :
0000 125 :
0000 126 :
0000 127 :
0000 128 :
0000 129 :
0000 130 :
0000 131 :
0000 132 :
0000 133 :
0000 134 :
0000 135 :
0000 136 :
0000 137 :
0000 138 :
0000 139 :
0000 140 :
0000 141 :
0000 142 :
0000 143 :
0000 144 :
0000 145 :
0000 146 :
0000 147 :
0000 148 :
0000 149 :
0000 150 :
0000 151 :
0000 152 :
0000 153 :
0000 154 :
0000 155 :
0000 156 :
0000 157 :
0000 158 :
0000 159 :
0000 160 :
0000 161 :
0000 162 :
0000 163 :
0000 164 :
0000 165 :
0000 166 :
0000 167 :
0000 168 :
0000 169 :
0000 170 :
0000 171 :
0000 172 :
0000 173 :
0000 174 :
0000 175 :
0000 176 :
0000 177 :
0000 178 :
0000 179 :
0000 180 :
0000 181 :
0000 182 :
0000 183 :
0000 184 :
0000 185 :
0000 186 :
0000 187 :
0000 188 :
0000 189 :
0000 190 :
0000 191 :
0000 192 :
0000 193 :
0000 194 :
0000 195 :
0000 196 :
0000 197 :
0000 198 :
0000 199 :
0000 200 :
0000 201 :
0000 202 :
0000 203 :
0000 204 :
0000 205 :
0000 206 :
0000 207 :
0000 208 :
0000 209 :
0000 210 :
0000 211 :
0000 212 :
0000 213 :
0000 214 :
0000 215 :
0000 216 :
0000 217 :
0000 218 :
0000 219 :
0000 220 :
0000 221 :
0000 222 :
0000 223 :
0000 224 :
0000 225 :
0000 226 :
0000 227 :
0000 228 :
0000 229 :
0000 230 :
0000 231 :
0000 232 :
0000 233 :
0000 234 :
0000 235 :
0000 236 :
0000 237 :
0000 238 :
0000 239 :
0000 240 :
0000 241 :
0000 242 :
0000 243 :
0000 244 :
0000 245 :
0000 246 :
0000 247 :
0000 248 :
0000 249 :
0000 250 :
0000 251 :
0000 252 :
0000 253 :
0000 254 :
0000 255 :
0000 256 :
0000 257 :
0000 258 :
0000 259 :
0000 260 :
0000 261 :
0000 262 :
0000 263 :
0000 264 :
0000 265 :
0000 266 :
0000 267 :
0000 268 :
0000 269 :
0000 270 :
0000 271 :
0000 272 :
0000 273 :
0000 274 :
0000 275 :
0000 276 :
0000 277 :
0000 278 :
0000 279 :
0000 280 :
0000 281 :
0000 282 :
0000 283 :
0000 284 :
0000 285 :
0000 286 :
0000 287 :
0000 288 :
0000 289 :
0000 290 :
0000 291 :
0000 292 :
0000 293 :
0000 294 :
0000 295 :
0000 296 :
0000 297 :
0000 298 :
0000 299 :
0000 300 :
0000 301 :
0000 302 :
0000 303 :
0000 304 :
0000 305 :
0000 306 :
0000 307 :
0000 308 :
0000 309 :
0000 310 :
0000 311 :
0000 312 :
0000 313 :
0000 314 :
0000 315 :
0000 316 :
0000 317 :
0000 318 :
0000 319 :
0000 320 :
0000 321 :
0000 322 :
0000 323 :
0000 324 :
0000 325 :
0000 326 :
0000 327 :
0000 328 :
0000 329 :
0000 330 :
0000 331 :
0000 332 :
0000 333 :
0000 334 :
0000 335 :
0000 336 :
0000 337 :
0000 338 :
0000 339 :
0000 340 :
0000 341 :
0000 342 :
0000 343 :
0000 344 :
0000 345 :
0000 346 :
0000 347 :
0000 348 :
0000 349 :
0000 350 :
0000 351 :
0000 352 :
0000 353 :
0000 354 :
0000 355 :
0000 356 :
0000 357 :
0000 358 :
0000 359 :
0000 360 :
0000 361 :
0000 362 :
0000 363 :
0000 364 :
0000 365 :
0000 366 :
0000 367 :
0000 368 :
0000 369 :
0000 370 :
0000 371 :
0000 372 :
0000 373 :
0000 374 :
0000 375 :
0000 376 :
0000 377 :
0000 378 :
0000 379 :
0000 380 :
0000 381 :
0000 382 :
0000 383 :
0000 384 :
0000 385 :
0000 386 :
0000 387 :
0000 388 :
0000 389 :
0000 390 :
0000 391 :
0000 392 :
0000 393 :
0000 394 :
0000 395 :
0000 396 :
0000 397 :
0000 398 :
0000 399 :
0000 400 :
0000 401 :
0000 402 :
0000 403 :
0000 404 :
0000 405 :
0000 406 :
0000 407 :
0000 408 :
0000 409 :
0000 410 :
0000 411 :
0000 412 :
0000 413 :
0000 414 :
0000 415 :
0000 416 :
0000 417 :
0000 418 :
0000 419 :
0000 420 :
0000 421 :
0000 422 :
0000 423 :
0000 424 :
0000 425 :
0000 426 :
0000 427 :
0000 428 :
0000 429 :
0000 430 :
0000 431 :
0000 432 :
0000 433 :
0000 434 :
0000 435 :
0000 436 :
0000 437 :
0000 438 :
0000 439 :
0000 440 :
0000 441 :
0000 442 :
0000 443 :
0000 444 :
0000 445 :
0000 446 :
0000 447 :
0000 448 :
0000 449 :
0000 450 :
0000 451 :
0000 452 :
0000 453 :
0000 454 :
0000 455 :
0000 456 :
0000 457 :
0000 458 :
0000 459 :
0000 460 :
0000 461 :
0000 462 :
0000 463 :
0000 464 :
0000 465 :
0000 466 :
0000 467 :
0000 468 :
0000 469 :
0000 470 :
0000 471 :
0000 472 :
0000 473 :
0000 474 :
0000 475 :
0000 476 :
0000 477 :
0000 478 :
0000 479 :
0000 480 :
0000 481 :
0000 482 :
0000 483 :
0000 484 :
0000 485 :
0000 486 :
0000 487 :
0000 488 :
0000 489 :
0000 490 :
0000 491 :
0000 492 :
0000 493 :
0000 494 :
0000 495 :
0000 496 :
0000 497 :
0000 498 :
0000 499 :
0000 500 :
0000 501 :
0000 502 :
0000 503 :
0000 504 :
0000 505 :
0000 506 :
0000 507 :
0000 508 :
0000 509 :
0000 510 :
0000 511 :
0000 512 :
0000 513 :
0000 514 :
0000 515 :
0000 516 :
0000 517 :
0000 518 :
0000 519 :
0000 520 :
0000 521 :
0000 522 :
0000 523 :
0000 524 :
0000 525 :
0000 526 :
0000 527 :
0000 528 :
0000 529 :
0000 530 :
0000 531 :
0000 532 :
0000 533 :
0000 534 :
0000 535 :
0000 536 :
0000 537 :
0000 538 :
0000 539 :
0000 540 :
0000 541 :
0000 542 :
0000 543 :
0000 544 :
0000 545 :
0000 546 :
0000 547 :
0000 548 :
0000 549 :
0000 550 :
0000 551 :
0000 552 :
0000 553 :
0000 554 :
0000 555 :
0000 556 :
0000 557 :
0000 558 :
0000 559 :
0000 560 :
0000 561 :
0000 562 :
0000 563 :
0000 564 :
0000 565 :
0000 566 :
0000 567 :
0000 568 :
0000 569 :
0000 570 :
0000 571 :
0000 572 :
0000 573 :
0000 574 :
0000 575 :
0000 576 :
0000 577 :
0000 578 :
0000 579 :
0000 580 :
0000 581 :
0000 582 :
0000 583 :
0000 584 :
0000 585 :
0000 586 :
0000 587 :
0000 588 :
0000 589 :
0000 590 :
0000 591 :
0000 592 :
0000 593 :
0000 594 :
0000 595 :
0000 596 :
0000 597 :
0000 598 :
0000 599 :
0000 600 :
0000 601 :
0000 602 :
0000 603 :
0000 604 :
0000 605 :
0000 606 :
0000 607 :
0000 608 :
0000 609 :
0000 610 :
0000 611 :
0000 612 :
0000 613 :
0000 614 :
0000 615 :
0000 616 :
0000 617 :
0000 618 :
0000 619 :
0000 620 :
0000 621 :
0000 622 :
0000 623 :
0000 624 :
0000 625 :
0000 626 :
0000 627 :
0000 628 :
0000 629 :
0000 630 :
0000 631 :
0000 632 :
0000 633 :
0000 634 :
0000 635 :
0000 636 :
0000 637 :
0000 638 :
0000 639 :
0000 640 :
0000 641 :
0000 642 :
0000 643 :
0000 644 :
0000 645 :
0000 646 :
0000 647 :
0000 648 :
0000 649 :
0000 650 :
0000 651 :
0000 652 :
0000 653 :
0000 654 :
0000 655 :
0000 656 :
0000 657 :
0000 658 :
0000 659 :
0000 660 :
0000 661 :
0000 662 :
0000 663 :
0000 664 :
0000

```

0006 107 .SBTTL Main routine
0006 108 ++
0006 109 :SERAPAT
0006 110
0006 111 Functional description:
0006 112
0006 113 In order to perform a multi-step security erase, the caller repeatedly
0006 114 calls this service, each time incrementing the iteration count. After
0006 115 each call, the erase pattern returned is written in the user supplied
0006 116 area. (The user is responsible for propagating that pattern throughout
0006 117 memory, disk, tape, etc.) When the service returns SSS_NOTRAN in R0,
0006 118 the security erase operation is complete.
0006 119
0006 120 This simple routine will always return an erase pattern of 0. It is
0006 121 up to the system manager to provide a specialized load algorithm.
0006 122
0006 123 Calling sequence:
0006 124
0006 125 This routine should be called via a CALLS/G to EXESERAPAT.
0006 126
0006 127 Input:
0006 128
0006 129 TYPE(AP) : Security erase type. The legal types are
0006 130 ERASK_MEMORY : main memory
0006 131 ERASK_DISK : disk storage
0006 132 ERASK_TAPE : tape storage
0006 133
0006 134 COUNT(AP) : Iteration count. The service should be called
0006 135 the first time with the value 1, then 2, etc.,
0006 136 until the status SSS_NOTRAN is returned. The
0006 137 local symbol MAXCOUNT defines how many times this
0006 138 happens.
0006 139
0006 140 Output:
0006 141
0006 142 PATADR(AP) : Address of a longword into which the security
0006 143 erase pattern is to be written.
0006 144
0006 145 Routine value:
0006 146
0006 147 R0 = SSS_ACCVIO : pattern output area not accessible
0006 148 SSS_BADPARAM : invalid security type code
0006 149 SSS_NORMAL : normal successful completion
0006 150 SSS_NOTRAN : security erase complete
0006 151 :--
0006 152
0006 153 .PSECT Y$EXEPAGED : This code is pageable
0006 154
0006 155 EXESERAPAT RTN::: : SERAPAT code
0006 156 MOVWL #SSS_BADPARAM,R0 : Assume bad parameter value
0006 157 MOVL TYPE(AP),R1 : Get the type code
0006 158 ASSUME ERASK_MINTYPE EQ 1 : This must be true if BLEQ is to work
0006 159 BLEQ 69$ : Branch if type code too small
0006 160 CMPL #ERASK_MAXTYPE,R1 : Is the type code too big?
0006 161 BLSS 69$ : Branch if yes
0006 162 MOVL COUNT(AP),R1 : Get the count
0006 163 BLEQ 69$ : Branch if too small

```

50 0629 8F	3C 0014	164	MOVZWL #SSS_NOTRAN, R0	: Assume count too big
51 01	D1 0019	165	CMPL #MAXCOUNT, R1	: Are we done?
12	19 001C	166	BLSS 69\$: If less, then yes
50 0C	3C 001E	167	MOVZWL #SSS_ACCVIO, R0	: Assume access violation
51 0C AC	D0 0021	168	MOVL PATAADR(AP), R1	: Get address of user buffer
50 61	0025	169	IFNOWRT #4, (R1), 69\$: Branch if no write access
50 01	D4 002B	170	CLRL (R1)	: Return the erase pattern
04	002D 0030	171	MOVZWL #SSS_NORMAL, R0	: Set success status
0031	172	69\$: RET		: Return
0031	173		.END	

```

COUNT          = 00000008
ERASK_MAXTYPE = 00000003
ERASK_MINTYPE = 00000001
EXE$ERAPAT    = 00000000 RG 02
EXE$ERAPAT_RTN = 00000000 RG 04
EXE$ERAPAT_VEC = 00000000 RG 03
MAXCOUNT      = 00000001
PATADR        = 0000000C
SSS_ACCVIO    = 0000000C
SSS_BADPARAM  = 00000014
SSS_NORMAL    = 00000001
SSS_NOTRAN   = 00000629
TYPE          = 00000004

```

! Psect synopsis !

PSECT name

	Allocation	PSECT No.	Attributes
ABS .	00000000	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
AEXENONPAGED	00000008	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
SSS\$500	00000006	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
YSEXEPAGED	00000031	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase

	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.08	00:00:00.75
Command processing	131	00:00:00.57	00:00:04.17
Pass 1	207	00:00:04.38	00:00:14.66
Symbol table sort	0	00:00:00.68	00:00:01.85
Pass 2	52	00:00:00.84	00:00:02.83
Symbol table output	3	00:00:00.03	00:00:00.03
Psect synopsis output	2	00:00:00.04	00:00:00.32
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	432	00:00:06.62	00:00:24.67

The working set limit was 1200 pages.

23328 bytes (46 pages) of virtual memory were used to buffer the intermediate code.

There were 30 pages of symbol table space allocated to hold 460 non-local and 1 local symbols.

174 source lines were read in Pass 1, producing 17 object records in Pass 2.

12 pages of virtual memory were used to define 11 macros.

! Macro library statistics !

Macro library name

-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2
TOTALS (all libraries)

Macros defined

1
7
8

533 GETS were required to define 8 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LISS:SYSERAPAT/OBJ=OBJ\$:SYSERAPAT MSRC\$:SYSERAPAT/UPDATE=(ENH\$:SYSERAPAT)+EXECMLS/LIB

0384 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SYSERAPAT
LIS

SYSFAO
LIS

SYSGETDVI
LIS

SYSEXIT
LIS

SYSEUTSRU
LIS

SYSFORCEX
LIS

SYSGETJPI
LIS